**Supplementary Material**

# Observer-independent quantification of insulin granule exocytosis and pre-exocytotic mobility by TIRF microscopy

Magnus Matz[1], Kirstin Schumacher[2], Kathrin Hatlapatka[2], Dirk Lorenz[3], Knut Baumann[1], Ingo Rustenbeck[2]

[1]Institute of Medicinal and Pharmaceutical Chemistry
[2]Institute of Pharmacology and Toxicology
[3]Institute of Analysis and Algebra, University of Braunschweig, Braunschweig, Germany

Please direct correspondence regarding the MATLAB code to: k.baumann@tu-bs.de

Please check http://www.pharmchem.tu-bs.de/forschung/baumann/ for updates and bug reports.

## Short documentation of the in-house written MATLAB code for the analysis of TIRF image sequences and illustration of the analysis of a sample sequence.

## Contents

**System requirements**

The code was developed and tested with Matlab version R2012a under Windows® 7.

Required Toolboxes:

- Statistics Toolbox
    - Functions called: *chi2inv* (beta_diffusion_cutoff.m), *nlinfit* (beta_CD_analysis), *pdist*, *linkage*, *cluster* (beta_flash_detect)
- Image Processing Toolbox
    - Functions called: *imfilter* (beta_start), *imshow* (beta_flash_detect)

Package contents:

M-Files:

beta_start.m (main script)
beta_options.m (script to set the options)
beta_clustering_options.m (additional options for advanced users)
beta_diffusion_cutoff.m (options to distinguish between diffusion and directed movement)

P-Files:

beta_CD_analysis.p
beta_background.p
beta_cd.p
beta_conv3.p
beta_evaluation.p
beta_flashdetect.p
beta_flashpos.p
beta_follow.p
beta_msd.p
beta_peakborders.p
beta_peakfind.p
beta_rename.p
beta_savgo_deriv.p
beta_smoothing_kernel.p
beta_smooth_no_toolbox.p
beta_sqeudist.p
beta_udetect.p
beta_utrack.p

# 1 Introduction

The evaluation program presented here extracts information from TIRF microscopic image sequences in three consecutive steps.

1. Granule detection
   - Counting all the objects
   - Presenting lists of the detected objects
   - Characterizing these objects by their size and intensity
2. Granule tracking
   - Calculation of statistical parameters like the MSD and the Caging Diameter
   - Connecting the tracked objects with their intensities to generate intensity-time-curves
3. Detection of exocytosis
   - Detection of pixels in the sequence with explicit intensity increase
   - Defining the dissipating cloud of fluorescent material and identifying the fused granule
   - Connecting the information on tracked objects with the fusion event

The program is provided as precompiled MATLAB files, together with a main script and several option files for customized settings.

Each sequence needs to be saved as images in a format supported by the MATLAB function *imread* (e.g. TIFF) in a separate folder. The required image depth is 12 bit. Images with a lower image depth may not be sufficient to describe exocytotic events adequately. All images should be named with a constant root (i.e. word stem), followed by an increasing number with a constant number of digits (i.e. use leading zeros to assure that the operating system reads the images consecutively). The sample sequence is named t=5__000.tiff, where „t=5__“ is the root and “000” to “199” enumerates the 200 images ordered according to acquisition time.

*beta_start* is the main script for the program, while *beta_options* lists the needed settings for the main routine. Please refer to *beta_options* for a documentation of all available options. *beta_clustering_options* and *beta_diffusion_cutoff* can be edited for customizing the clustering routine and customizing the calculation of the cutoff for the estimation of directed movement (as this depends of the temperature for example).

The folder holding the MATLAB files needs to be on your MATLAB path. Change the current folder to the folder holding the image files and run *beta_start*. Granule detection, granule tracking and exocytosis detection is now executed for the sequence in the current folder. Additional analyses like the diffusion cutoff for the estimation of directed movement and an advanced analysis of the Caging Diameter are not routinely executed. To use them, the return statement on line 172 of *beta_start* needs to be removed.

The program displays the progress of the analysis. In addition to that, several figures are plotted and the results of the analysis are stored in the workspace.

# 2 General steps

The images are read from the files and are saved in the MATLAB workspace in three matrices, each starting with *Data_*. The images can be cut prior to the analysis to evaluate only a specific part of each image. The rectangle defining the actually analysed area can be set in *beta_options*. The raw data are stored in *Data_backup*. *Data_flash* stores images with specific information, showing solely the pixels with sharply increasing intensity. *Data_flash_clustered* contains the clustered information of these pixels and shows the detected dissipating clouds. In a fourth matrix, the smoothed Data is stored (*Data_smooth*). This matrix is deleted after the object detection step. If the matrix shall be kept, *Data_smooth* needs to be deleted from the *clear* statement on line 145 of *beta_start*.

The variables *m* and *n* give the dimensions of the images, while *o* gives the number of images in the sequence.

In addition to the preparatory steps, the background intensity is estimated. The result, which is based on finding the mode in the intensity histogram, is plotted by default (see Figure 1). The blue line indicates the occurrence of the respective intensity values of all pixels in the evaluated sequence, while the red curve represents its first derivative. The background intensity is set to the intensity level corresponding to the maximum in the intensity histogram (i.e. the intensity value that appears most often). For the sample sequence, the intensity level of the background is set to 55.1.

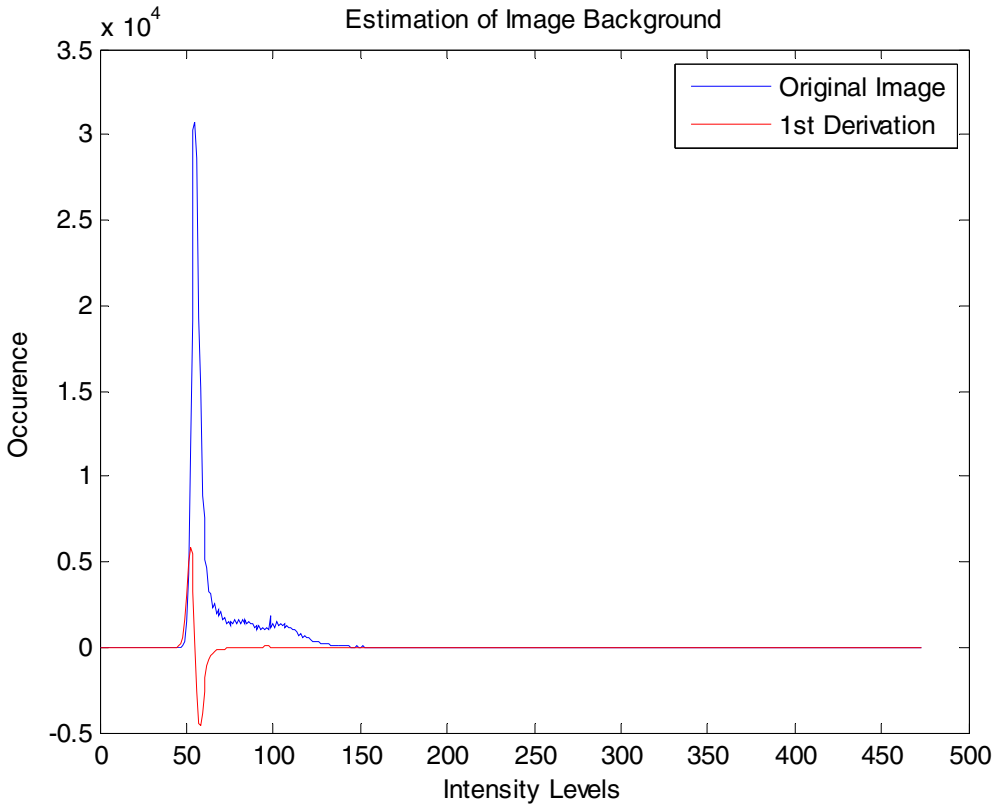


Figure 1: Estimation of the background intensity level in the sample sequence. The background intensity is set to the intensity level corresponding to the maximum in the intensity histogram. It is determined as the zero-crossing of the smoothed 1st derivation of the histogram.

# 3 Granule detection

The number of granules in each image is listed in the vector *objectcount* of length *o* (i.e. the number of images). Hence, the $i^{th}$ value in *objectcount* represents the number of detected granules in the $i^{th}$ image. The graph can be plotted by typing *plot(objectcount)* (see Figure 2).

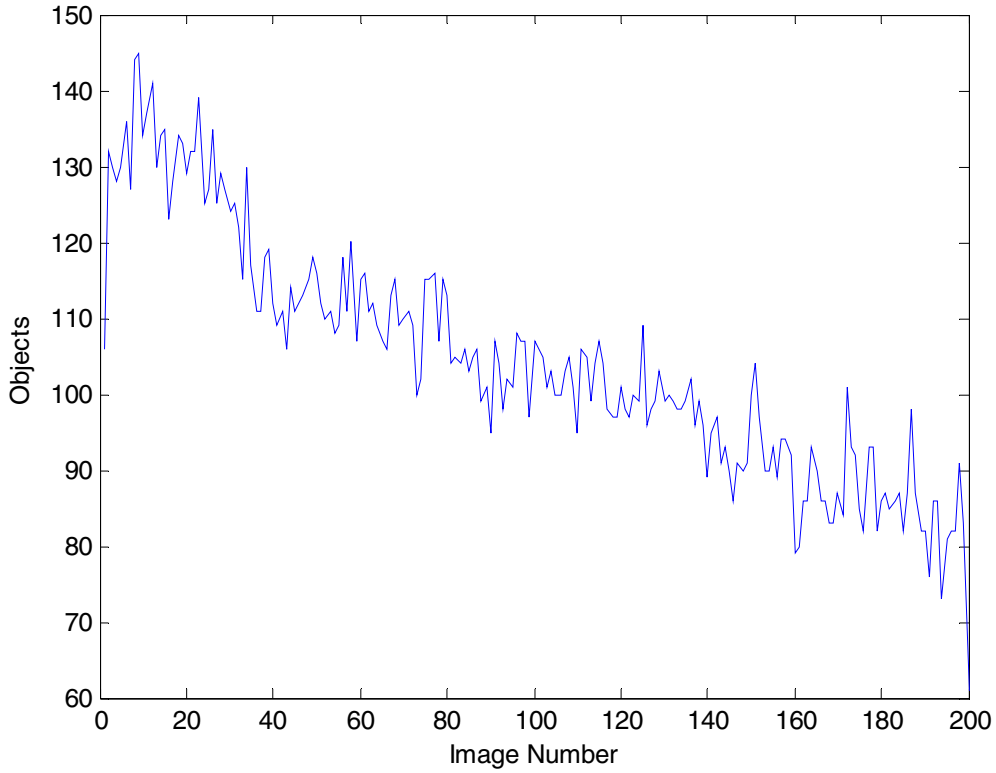


Figure 2: Number of objects versus the image number of the processed sequence. The plot indicates a reduction of visible objects over time.

The detected objects in the sequence are listed in *objects.* This is a 3D array. It has the dimensions *max_objects*, *9, o*. Here, *max_objects* describes the maximum number of detected objects in a single image in the sequence, corresponding to *max(objectcount).* In images with less than *max_objects*, the array is filled with zeros for the positions *objectcount(i)*+1 to *max_objects* in the $2^{nd}$ dimension of the array.

The nine columns of *objects* describe the following object characteristics:

1. X-Coordinate of the object's intensity maximum
2. Y-Coordinate of the object's intensity maximum
3. Background-corrected maximum intensity
4. Background-corrected mean maximum intensity (3x3 mean centred at object's maximum intensity)
5. Base area intensity
6. Base area-corrected intensity (i.e. maximum intensity minus base area intensity)
7. Base area-corrected mean intensity (3x3 mean)
8. Raw mean intensity in backup images (3x3 mean)
9. Base Radius

# 4 Granule tracking

For tracking the granules, several characteristics are needed. These are specified with the variable *maximumrows.* These characteristics include the coordinates of the granules (columns 1 and 2 in *objects*), as well as the base radius (column 9) and one of the intensity values (columns 3, 4, 6, 7, 8). For tracking the raw mean intensity (column 8) performed best and is used as default value. Hence, the default object characteristics for tracking can be found in columns [1, 2, 8, 9] of array *objects* and these are passed to the respective tracking functions with variable *maximumrows* on line 153 in *beta_start*.

After the tracking, an overview over the trajectories is generated and all trajectories of objects with a minimum duration time are plotted. The minimum duration time for this plot can be edited in *beta_options* with the variable *options.tracking_display_limit*. For the sample sequence, the plot shown in Figure 3 results. The plot shows the tracks with the predefined minimum duration time and their respective Tracking-IDs. Zooming into the plot allows a detailed visual analysis.

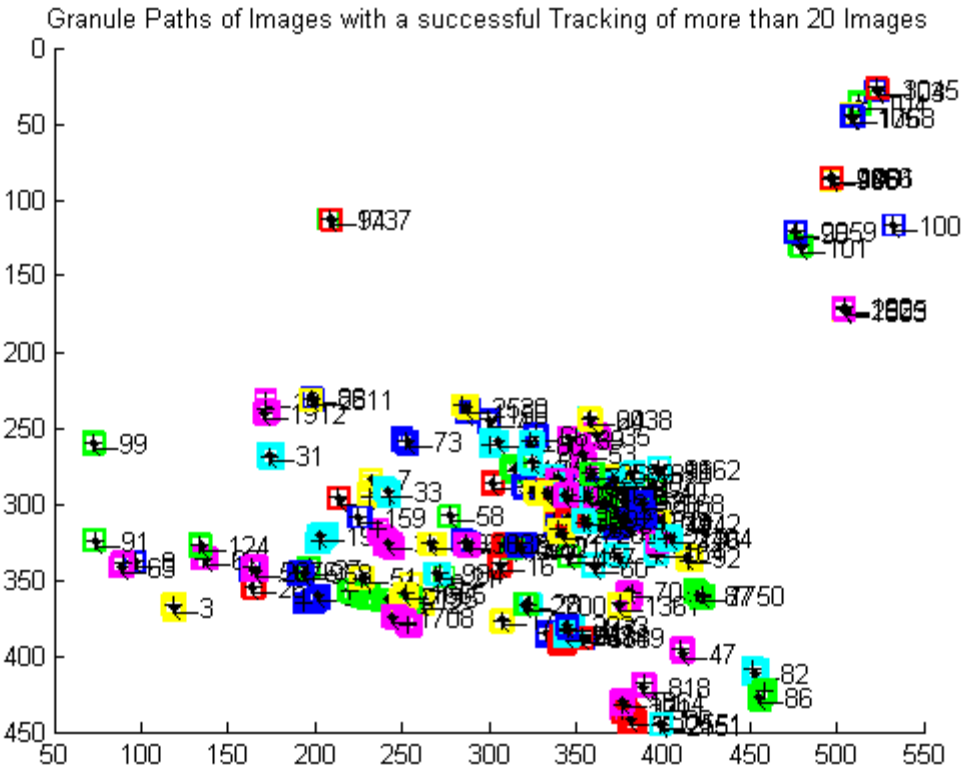


Figure 3: Granule paths of the sample sequence. All trajectories with a minimum duration time of more than 20 time points (i.e. images) are displayed. In addition, the tracking-ID is displayed for each trajectory.

During the tracking of the granules, the matrix *indexlist* is created (see below). With this matrix, each object in *objects* can be tracked through all images by its tracking-ID. *indexlist* is a matrix with the dimensions [*max_objects*, *o*]. In column *i*, the $j^{th}$ entry represents the $j^{th}$ object in image *i* (as listed in *objects*). The respective entry in this cell represents the Tracking-ID.

The table below shows a part of the matrix *indexlist* for the sample sequence. The first granule in the first image is linked to the first granule of the second image. Its Tracking-ID is 1. The third granule of the first image is linked to the fourth granule in the second image. Its Tracking-ID is 3. From this information a trajectory can be generated by drawing a path along the object's coordinates with the same Tracking-ID (see Figure 3). The trajectories can be evaluated by calculating the Mean Squared Displacement (stored in *msd*) or the Caging Diameter (stored in *diameter*). The $i^{th}$ row of the matrices *msd* and *diameter* contains the information of the $i^{th}$ trajectory (i.e. the trajectory with Tracking-ID i).

Image Number

| Object Number | **1** | **2** | **3** |
|---|---|---|---|
| **1** | 1 | 1 | .. |
| **2** | 2 | 2 | .. |
| **3** | 3 | 4 | .. |
| **4** | 4 | 3 | .. |
| **5** | 5 | 5 | .. |
| **6** | 6 | 8 | .. |
| **7** | .. | .. | .. |

Extract of *indexlist* after evaluating the example sequence

Moreover, the intensities of all granules with the same Tracking-ID are extracted from *objects* to produce intensity profiles which are stored in matrix *intensities.* The dimension of *intensities* and *diameter* are [*max(Tracking_ID), o*], the dimension of *msd* is [*max(Tracking_ID), o-1*]. In matrix *intensitites*, zero values represent images where the respective granule is not present. In matrices *diameter* and *msd*, zero values may also be zero values of these parameters (i.e. no movement). Differentiating "no movement" from "object is not present in that image" in matrices *msd* and *diameter* must be done by checking the respective value in matrix *intensities* in addition to *msd* or *diameter*, respectively.

As mentioned before, the information contained in the trajectories can be visualized either as path in the x-y plane (cf. Figure 3) or as an intensity profile. The intensity profile of the trajectory with the Tracking-ID 2 is shown in Figure 4. The granule is present from the very first image and disappears in image 128. The intensity increases sharply in image 19 owing to exocytosis (see Section 5).

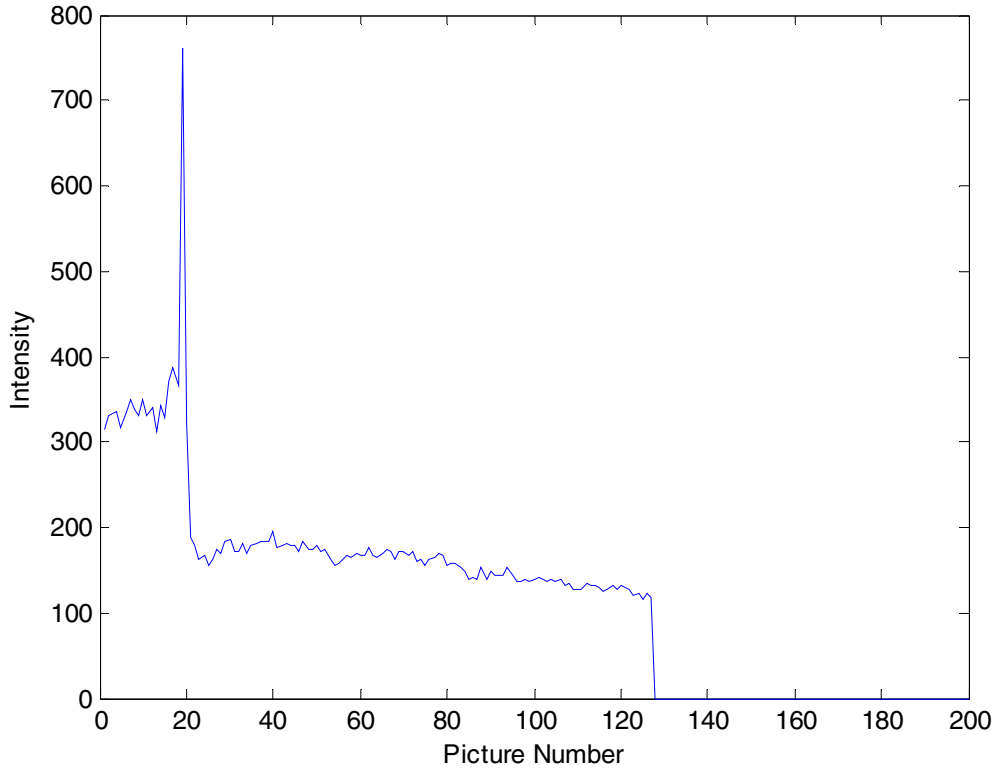


Figure 4: Intensity profile of the trajectory with the Tracking-ID 2. The granule is present from the first image till image 127. Then it disappears. The intensity increases sharply in image 19.

The matrix *analysis* shows the basic characteristics of the trajectories. The dimension of *analysis* is [*max(Tracking_ID), 3*]. The rows indicate the *Tracking-ID* of the different trajectories, the columns show the total moved distance of the granule during the tracking, the Euclidean distance from the start to the end of the trajectory and the residence time of the granule in the evanescent field.

# 5 Detection of exocytosis

After tracking the granules, the detection of exocytotic events is started. In this step, pixels with a sharply increasing intensity are identified and clustered.

The following images show the processed image 19 where Figure 5 shows the 19$^{th}$ image of *Data_flash*, while Figure 6 shows the 19$^{th}$ image of *Data_flash_clustered*.

A bright spot in the lower part of the cell can be seen. This bright spot –representing an exocytosis event– is an accumulation of multiple pixels with a simultaneously increasing intensity and is correctly identified as an exocytosis event with the algorithm detailed in the main paper.

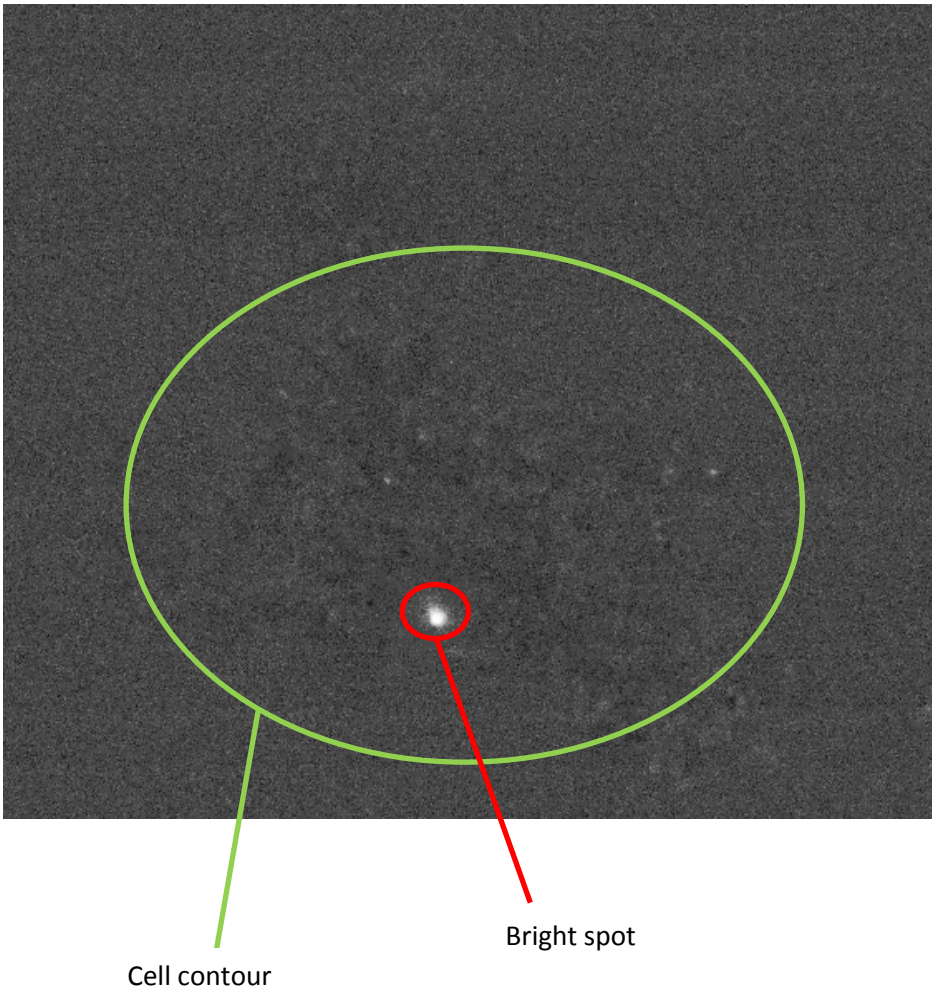


Figure 5: Image 19 of array *Data_flash*. Many neighboring pixels with sharply increasing intensity can be seen.

In the following, all pixels with intensities below the threshold value *Flash_limit*, which can be set in *beta_clustering_options*, are set to 0. The remaining pixels are clustered with a hierarchical clustering in order to decide whether or not they belong to the dissipating cloud of fluorescent material released by exocytosis. The distance cutoff for a successful clustering is set in *beta_clustering_options* with the variable *MaxPixelDistance*. Clusters with lesser pixels than defined in *MinClusterSize* are deleted. All remaining clusters are identified as an exocytotic event (see Figure 6). The default cutoffs were determined and validated with the available sequences. In *beta_clustering_options* alternative cutoffs are proposed for different camera systems. These values were determined empirically in cooperation with other research groups.

Figure 6: Only one cluster of pixels with sharply increasing intensities remains. Smaller clusters and isolated pixels were deleted since they were either generated by granules moving in z-direction or by background noise.

In the following, granules inside the clustered area shortly before the exocytosis event are located and it is checked which of them would most likely be the trigger of the exocytotic event. This decision is based on the position of the granule within the cluster (distance to the centre) and the intensity of profiles of the candidate granules (see below).

All detected exocytotic events of the evaluated sequence are listed in *allflashes*. The table below shows *allflashes* for the sample sequence (the first row is inserted in this manual for illustrative purpose).

| Image No. | Area | ID 1 | ID 2 | ID3 |
|---|---|---|---|---|
| 19 | 2912 | 2 | 2 | 2 |
| 40 | 2470 | 3 | 3 | 3 |
| 171 | 1222 | 10 | 10 | 10 |
| 156 | 251 | 61 | 61 | 61 |
| 68 | 178 | 44 | 44 | 44 |

The matrix *allflashes*, obtained from the evaluation of the sample sequence.

It can be seen that five exocytotic events are identified. The events occur in the images listed in the first column. The area of the dissipating cloud is shown in the second column. The exocytotic events in *allflashes* are sorted in descending order by this value, assuming that the larger the dissipating cloud is the more likely the identification of exocytosis gets. With small dissipating clouds, there is a growing probability that the arrival of a granule in the evanescent field is mistaken for an exocytotic event. The IDs listed in the following columns in *allflashes* are the Tracking-IDs of the candidate granules within the dissipating cloud which fulfill the excocytosis detection criteria as described in the accompanying paper. If all three criteria point to one and the same granule the identification is fairly safe.

For each excytotic event, the image with the exocytotic event and the granules with the Tracking-IDs is plotted with the image number as title.

Exocytosis detection criteria for granules within the dissipating cloud. The colors refer to the marker color for the granule that most likely led to the exocytosis event in respective figure (see Figure 7)

- ID 1: Optimal Distance to Center (red)
- ID 2: Optimal I1 (blue)[1]
- ID 3: Optimal I2 (green)[1]

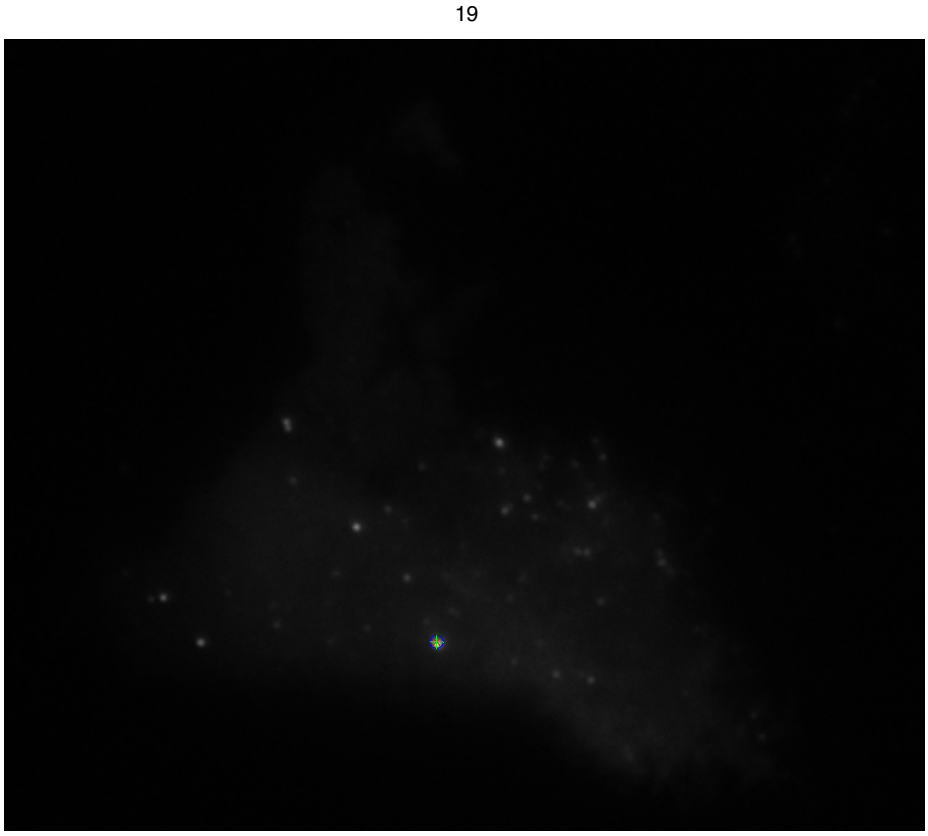


Figure 7: This image shows image number 19 and the detected source granule of the exocytosis. It is marked and its tracking-ID can be found in matrix *allflashes*.

[1] The definition of I1 and I2 can be found in the paper. They are based on a sharp increase of the intensity profile (I1) and a sharp decrease of the intensity profile (I2).

# 6 Additional Analysis

As mentioned in the introduction, additional analysis tools are implemented, but they are not activated in the basic settings. In *beta_start*, two additional scripts can be executed by removing the the *return* command on line 172.

With the first script, *beta_diffusion_cutoff*, a threshold for the caging diameter to differentiate between diffusion and directed movement can be calculated. Several parameters such as the average radius of the observed granules, the temperature and the radiography rate must be provided. The script then calculates the desired threshold and it is saved in *Caging_Diameter_Limit*. For the sample sequence and a CD_interval of 9 time points, a threshold of 2.8945 pixel / CD_interval is calculated.

With the second script, *beta_CD_analysis*, an additional analysis of the caging diameter is possible. This analysis is further described in Nofal, S., Becherer, U., Hof, D., Matti, U., & Rettig, J. (2007), *Primed Vesicles Can Be Distinguished from Docked Vesicles by Analyzing Their Mobility,* The Journal of Neuroscience, 27(6), 1386–1395. In short, a parameter for the characterization of the movement of the granule population of the entire sequence is calculated. This is done with the help of a cumulative caging diameter function. The half-maximum-value of this function is used to characterize the granule population. Its unit is equal to the caging diameter and is pixels / CD_interval. Large half-maximal-values indicate a fast moving population, while low half-maximal-values indicate a slow moving population.